

# MINOS: Steps Towards a Syntax-free Computer Algebra Program

[Kyril Tintarev](#)

*Dept. of Mathematics, Uppsala University, Sweden*

The idea of Mathematical Interface with Natural Object Syntax, or *MINOS* for short, was a product of the author's increasing concern about the shortcomings of leading symbolic calculus software, such as *Maple* or *Mathematica*, as a tool for instruction of mathematics. After running several *Maple*-assisted laboratories for calculus and algebra, first at University of California, Irvine, and then at Uppsala, he arrived at a viewpoint that no pedagogic effort can focus the freshman's attention away from the intricacies of the command syntax and back to the mathematical content, so there is an urgent need for a more user-friendly software package. Even regular users of *Maple* have difficulty remembering most of its commands accurately. Ask a mathematics teacher after the summer vacation what *Maple* command draws a circle as a curve  $t \rightarrow (\cos(t), \sin(t))$  and you will probably be told to look up the manual.

It is not enough to replace the command line interface, like *Derive* does, by menus (that point to the same old commands) and uniform dialog boxes (for input of arguments of the commands). Although this protects the user from typing errors in the keywords and saves him from remembering the sequences of brackets, braces, commas, colons and semicolons that follow the keyword in the command line interfaces, the treatment of mathematical objects by the existent menu-dialog interfaces still imposes on the user a non-mathematical semantic environment. The author has also found that the record of computations in the form of a log also forces on the user non-mathematical semantics. Consider for example the two following lines:

```
x = a+2  
a = 3
```

For a mathematician they mean two equations that one is free to interpret as a system or as two separate statements. However, for a typical mathematical software package these lines represent two assignment operators, and after they are executed the variable  $x$  either contains the value 5 or resides in a state of "suspended evaluation" and yields 5 after one types in something like *eval(x)*. Add another line,

```
x = sin(x)
```

and it becomes a recurrent assignment that makes  $x$  equal  $\sin(5)$ .

We rest our case on a belief that computer aided instruction of mathematics should not put a first year student into this alien environment.

This is not the opinion that most proponents of computer aided mathematics instruction hold today. The prevailing opinion says that the syntax of *Maple* or *Mathematica* is something a technical student has to learn anyway, that these programs are here to stay as a part of the new scientific and engineering culture, together with *Emacs* and *TeX*, and so the effort spent by a mathematics instructor on teaching the command syntax of *Maple* alongside mathematics is

effort wisely spent. However, it has often happened that consensus lasts only as long as the current level of functionality is the best one available.

Our objection to the existent high-end mathematical programs is restricted only to the first year or two of college instruction. After a couple of years most students become proficient enough in programming in general to master *Mathematica* or *Maple* as their fifth or so computer language, and may use them for projects in, say, differential equations or Fourier analysis.

The underlying idea of *MINOS* is to take full advantage of object oriented programming and populate the screen with visual mathematical objects with robust behavior that matches the way a mathematician relates to a mathematical object. *MINOS* employs Microsoft's metaphor of a file system (the technical name is *ListView ActiveX* control), but it uses the standard 32x32 icons with a name box underneath, not for handling files, but for representation of mathematical variables, vectors, matrices, sequences and whatever else might be beneficial for a mathematician to see as a conceptional unit. Figure 1 shows a screenshot of *MINOS* (draft 009) with function  $f(x,y) = \sin(x*y)$  and its two partial derivatives, calculated by pressing the differentiation button; one of the derivatives is plotted. Operations with visual objects (we tentatively call a *MINOS* object *eidolon*) are performed with a mouse, for example by highlighting the icons of arguments or by dragging one of them onto another. Each object type has its own context menu with the main operations specific to this object (a matrix should have a determinant on its menu) as well as a few common ones (delete, copy and paste). Plotting is one of the easiest accessible features of *MINOS*: marking a check-box near an *eidolon* results in an automatic plotting, whenever the *eidolon* has a graphic representation. Plot types can be changed and the range of arguments readjusted by self-explanatory buttons and boxes on the graphics window. Figure 2 shows an overlay of graphs for a function  $f(x) = x \sin(x)$  and its derivative. Several functions dependent on the same variable can be graphed in the same axes merely by marking the check boxes.

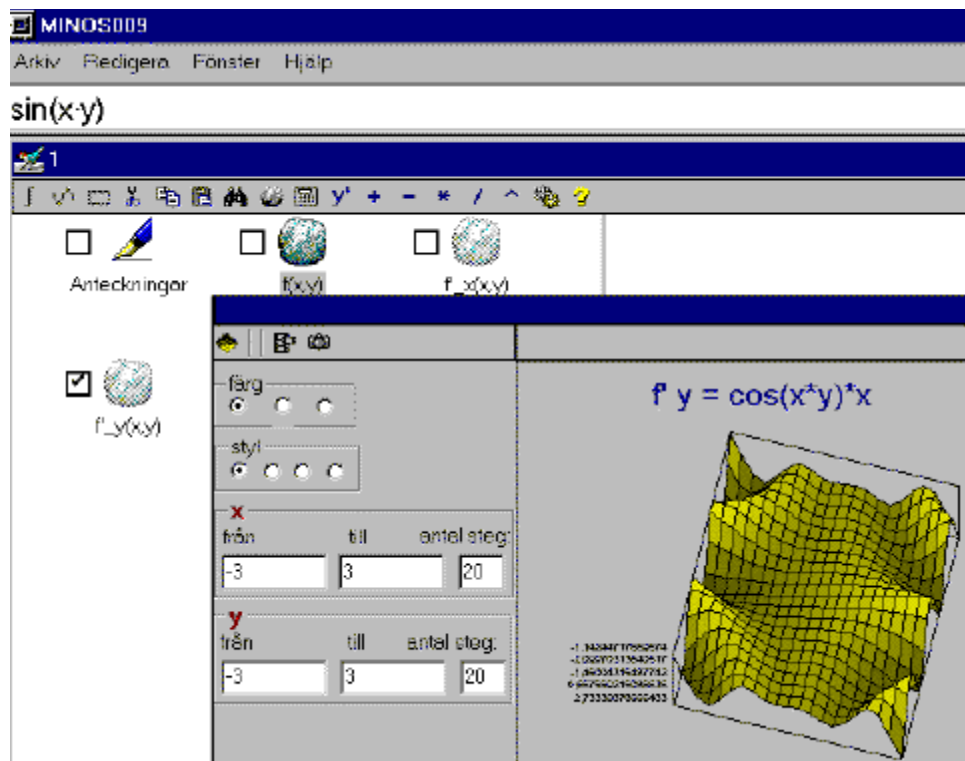
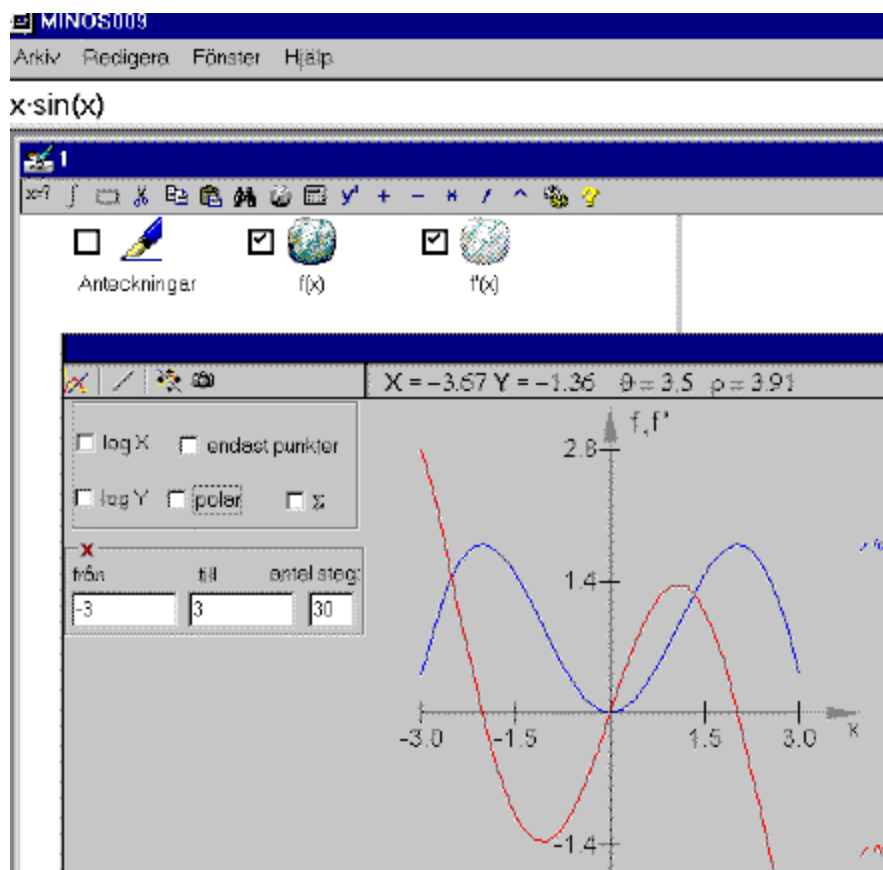


Figure 1. A screenshot of *MINOS* (draft 009) with function  $f(x,y) = \sin(x*y)$  and its two partial derivatives, calculated by pressing the differentiation button; one of the derivatives is plotted



**Figure 2. An overlay of graphs for a function  $f(x) = x \sin(x)$  and its derivative**

The main benefit of the new interface is a level of fluency - the student is now free from the concern of "how shall I do it?" and can better concentrate on the question "what does it mean?". Representation of the workspace as a log is abandoned: pedagogic research shows that the sequential approach to problems is inferior to the structural approach, and *MINOS* provides the latter. One enters the problem's data in the form of eidola, one creates other eidola with relations that seem relevant to the problem, and performs the calculations that output further eidola which can be stored in the same workspace. There is no chronological dependence between eidola: in our example above eidola  $x=a+2$ ,  $a=3$  and  $x=\sin(x)$  coexist without affecting each other. One can substitute  $a=3$  into  $x=a+2$  by dragging one icon onto the other, and one has to decide oneself if  $x=\sin(x)$  is to mean an equation or an iterative assignment. A finer tuning between the interface and the needs of instruction will still be needed: although the software is tested as it is written, most teachers prefer to evaluate only a completed work. *MINOS* is not meant to be a pedagogic solution by itself, only a better platform that allows teachers to develop more efficient laboratory sessions and project assignments.

This organization requires that the user executes commands step-by-step, not as program code (for program writing we find *Maple* and *Mathematica* less objectionable). We found this limitation bearable, as very few instructors of calculus or linear algebra give programming assignments in the freshmen computer laboratory sessions.

*MINOS* is expected to be used also as a presentation tool for mathematical lectures: it can be used dynamically to discuss students' spontaneous suggestions and examples, while existent programs are hard to use in class in a spontaneous manner - advance preparation is recommended.

A tentative prototype of *MINOS* was developed at Uppsala in 1997-1998 with a small intramural grant from the Unit for Instructional Development. Its interface was employing *MathEdge* (a developer edition of *Maple V, release 3*) for all computational tasks. However, the program was aborted because of licensing problems: Waterloo Maple Software has discontinued *MathEdge* and never offered us a license that would allow us to use it on a regular basis in a laboratory, not to say, to distribute *MINOS* elsewhere.

The project took an unpredictable turn: keeping the word "interface" in the name, we decided to rebuild *MINOS* completely as a program with its own computation engine. This was not without benefit, as instead of the string-exchange API (Application Program Interface) offered by *Maple*'s computation engine, *MINOS* integrates data structures of its computation engine into the visual mathematical objects of the interface as interoperable COM objects. At this risky stage financing of the project switched to private sources. Since the fall of 1998 *MINOS* has been supported by a grant of SKR 700 000 from The Foundation for Knowledge and Competence Development (KKS) matching the equal amount of private investment.

Under the terms of the sponsorship the first release of *MINOS*, the School Edition, due in February 2000, will be aimed at the high schools, covering the Swedish national curriculum for mathematics. The School Edition will be capable of algebraic manipulations of expressions, equation solving (mostly numerical, with symbolic solutions limited to equations used at school) integration and differentiation, vector algebra, a variety of 2D and 3D plotting modes, including animated plots, and elements of differential equations.

At the same time, the School Edition of *MINOS* is being developed as a platform capable of supporting add-on programs that are necessary for the university courses of calculus and linear algebra. Under favourable circumstances a college edition of *MINOS* may be finished within months after the release of the School Edition. It is not unlikely that *MINOS* will be used on both sides of the school-college divide as well as in adult and distance education. Building in an "object mail" capability, so that several participants could collaborate on the same mathematical assignment via Internet, is a part of the *MINOS* project as well.

The interface of *MINOS* has been tested with several groups of last year high school students. The following pattern has emerged: teachers opt for a five minutes introduction of the interface, and students need an average of fifteen minutes to handle *MINOS* with confidence (add-on libraries might need additional five minutes each). Students with time left eagerly engage in experimentation on their own and, when the teacher permitted, productively collaborated with each other on solving the problems. The testing at schools will continue in the fall 1999, and in 2000 we will begin testing *MINOS* in university instruction as well.

Development versions of *MINOS* are available to educational institutions upon registration with Eidonome, the commercial arm of the *MINOS* project, without cost. The company's address is

<http://www.eidonome.com/> Upon receiving the password one can download the current *MINOS* distribution, a 5MB zip file over a fast connection (a modem connection, presumably from a residence, will be timed out). This arrangement remains in force until the beginning of the beta testing program in November.

English and Swedish versions of *MINOS* are being developed and will be released simultaneously.

Kyril Tintarev  
Dept. of Mathematics  
Uppsala University  
Box 480  
S-751 06 Uppsala  
Sweden  
[kyril@math.uu.se](mailto:kyril@math.uu.se)